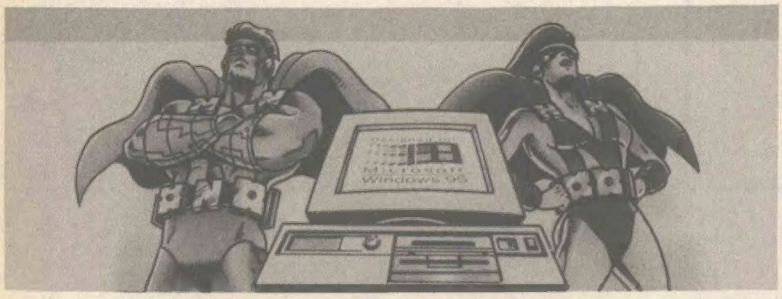
INTERNET

อรรคชัย กัมลาศพิทักษ์

JavaScript VS. Perl



เปรียบเทียบคุณสมบัติและความสามารถในการทำงานของ JavaScript VS. Perl

ถ้าพูดถึง JavaScript และ Perl เชื่อว่านักท่องอินเทอร์เน็ตส่วนใหญ่คงจะรู้จักและคุ้นเคยกันเป็นอย่างดี ไม่ว่าจะไปที่เว็บไหน ก็มักจะพบว่ามีการใช้งานไม่ภาษาใดก็ภาษาหนึ่ง บางแห่งก็มีการใช้งานทั้ง 2 ภาษา ด้วยความที่ทั้งคู่ต่างก็เป็นเครื่องมือที่ได้รับความนิยมในการใช้งานในเว็บไซท์ต่าง ๆ บนอินเทอร์เน็ต ทำให้มีคำถามตามมาเสมอว่า ระหว่าง JavaScript กับ Perl ใครเป็นภาษาที่ดีกว่ากัน? ใครได้รับความนิยมมากกว่าใคร? แล้วเราจะเลือกใช้ภาษาใดดี? แม้ทั้งคู่จะถูกใช้งานอย่างแพร่หลาย และมีความสามารถที่คล้ายคลึงกันหลาย ๆ ประการ แต่ก็มีจุดประสงค์ในการสร้างและการใช้งานที่แตกต่างกัน รวมถึงจุดเด่น-จุดด้อยที่ไม่เหมือนกันด้วย บทความนี้จะทำการเปรียบเทียบคุณสมบัติและความสามารถในการทำงานของทั้ง 2 ภาษา เพื่อช่วยให้ท่านที่กำลังพิจารณาที่จะศึกษาภาษาใดภาษาหนึ่ง หรือกำลังจะเลือกใช้งานกับเว็บไซท์ของตนเอง สามารถเลือกใช้ให้ตรงกับงานที่ต้องการต่อไป

รู้จักกับแต่ละภาษาก่อน

ก่อนจะดูคุณสมบัติเปรียบเทียบของ JavaScript และ Perl เราไปทำความรู้จักกับทั้งสองภาษากันก่อน อาจช่วยให้ได้รับทราบแนวคิดของการสร้างสรรค์แต่ละภาษาขึ้นมา และช่วยให้สามารถเลือกใช้งานได้ตามคุณลักษณะของแต่ละภาษาต่อไป

Perl

Perl ย่อมาจากคำว่า Practical Extraction and Report Language สร้างขึ้นโดย Larry Wall เพื่อช่วยงานเขียนโปรแกรมในระบบ UNIX โดยกำเนิดในปี ค.ศ. 1986 เมื่อ Larry ทำหน้าที่เป็นโปรแกรมเมอร์ในระบบ UNIX เขาจำเป็นต้องจัดการกับแฟ้มเอกสารที่อยู่ในรูปแฟ้มตัวอักษร (Text File) จำนวนมากที่มีความเกี่ยวโยงกัน เอกสารเหล่านี้สร้างความยุ่งยากให้กับเขาเป็นอย่างมาก จึงได้พยายามหาวิธีจัดการกับแฟ้มข้อมูลดังกล่าวโดยเริ่มจากการใช้ awk มาช่วยงาน แต่พบในเวลาต่อมาว่า awk ไม่สามารถตอบสนองความต้องการของเขาได้มากนัก เขาจึงได้เริ่มต้นพัฒนาโปรแกรมเพื่อช่วยงานของเขาขึ้นมาเอง โดยเขียนเป็นโปรแกรมยูทิลิตี้เพื่อช่วยจัดการกับเอกสารต่าง ๆ เช่นเดียวกับที่โปรแกรมเมอร์ในระบบ UNIX ทั่ว ๆ ไปทำกัน หลังจากนั้นไม่นานเขาก็พัฒนาขึ้นเป็นภาษาใหม่ โดยเขียนเป็น Interpreter เพื่อใช้ในการจัดการกับเอกสารที่อยู่ในรูปของตัวอักษรโดยเฉพาะ ต่อมาได้พัฒนาให้มีความสามารถเพิ่มขึ้น โดยการเพิ่มความสามารถในการกำหนดเงื่อนไขต่าง ๆ เช่นเดียวกับที่มีในภาษาโปรแกรมอื่น ๆ รวมถึงความสามารถในการทำงานบนเครือข่ายคอมพิวเตอร์ ภาษานี้ต่อมารู้จักกันในนามของ Perl และเป็นที่นิยมใช้งานกันอย่างรวดเร็วในหมู่โปรแกรมเมอร์ในระบบ UNIX ที่ต้องการความสะดวกรวดเร็วและง่ายในการเขียนโปรแกรม

Perl เลียนแบบเครื่องมือหลายอย่างใน sed และ awk แต่พัฒนาให้ทำงานได้ดีกว่า Perl เขียนขึ้นด้วยภาษา C แต่รูปแบบภาษาสั้นและกระชับกว่าภาษา C มาก Perl เป็นภาษาที่มีความยืดหยุ่นสูง เช่นเดียวกับภาษาระดับสูงทั่วไป Perl ยืมคุณสมบัติหลายอย่างจากภาษา C มาใช้ Perl ทำงานคล้าย Shell Script โดยเขียนโปรแกรมแล้วสั่งให้ตัวแปลภาษา Perl ทำการแปลแล้วทำงานตามคำสั่งนั้น ๆ ได้ทันที ทำให้การพัฒนาโปรแกรมประยุกต์ทำได้ง่ายและรวดเร็ว ที่สำคัญ Perl แจกจ่ายให้ใช้งานฟรี และมีเวอร์ชันที่ทำงานในเกือบทุกระบบปฏิบัติการ ไม่ว่าจะเป็น UNIX, Windows, Linux หรือ Macintosh สำหรับผู้ที่สนใจสามารถศึกษารายละเอียดเพิ่มเติมเกี่ยวกับ Perl รวมถึงดาวน์โหลดโปรแกรมมาติดตั้งใช้งานได้จาก

http://www.perl.com/ หรือ http://language.perl.com/

เมื่อ WWW ได้รับความนิยมอย่างมากจากผู้ใช้อินเทอร์เน็ต และมีการเปิดช่องทางให้ผู้ใช้งานสามารถเรียกใช้โปรแกรมประยุกต์บนเซอร์ฟเวอร์ได้โดยผ่านทาง CGI ของโปรแกรมที่ทำหน้าที่เป็นเว็บเซอร์ฟเวอร์ Perl ซึ่งถูกสร้างขึ้นเพื่อดำเนินการกับเอกสารที่เป็นตัวอักษรโดยเฉพาะ จึงเหมาะกับเอกสารแบบ HTML ซึ่งเป็นเอกสารแบบตัวอักษรด้วยเช่นกัน รวมถึงการที่ Perl เป็นอินเตอร์พรีตเตอร์ จึงไม่ต้องผ่านขั้นตอนการคอมไพล์โปรแกรมที่ยุ่งยากอย่างการใช้ภาษาอื่น ทำให้ง่ายในการใช้งาน Perl จึงกลายเป็นภาษายอดนิยมบนเครือข่ายอินเทอร์เน็ตอย่างรวดเร็ว มีการพัฒนาโปรแกรมประยุกต์ต่าง ๆ ขึ้นใช้งานจำนวนมากด้วย Perl และมีการแจกจ่ายโปรแกรมเหล่านี้ให้ใช้งานฟรีด้วย ไม่ว่าจะเป็นโปรแกรมสำหรับกรอกแบบฟอร์มต่าง ๆ โปรแกรมนับจำนวนผู้มาใช้บริการของเว็บไซท์ โปรแกรมค้นหาเอกสาร และอื่น ๆ อีกมากมาย

awk เป็นภาษาโปรแกรมที่พัฒนาขึ้นโดย AT&T ซึ่งจะมีมากับระบบปฏิบัติการ UNIX มีคำสั่งง่าย ๆ ที่ไม่สลับซับซ้อนนัก รวมทั้ง Built-in ฟังก์ชันจำนวนหนึ่งให้เรียกใช้งานได้ มักใช้จัดการกับข้อมูลที่เป็นตัวอักษร ในลักษณะที่คล้ายกับฐานข้อมูล โดยแต่ละบรรทัดของข้อมูลจะแทนแต่ละเรคอร์ด แบ่งแยกฟิลด์แต่ละฟิลด์ด้วยช่องว่างหรือเครื่องหมายอื่นก็ได้ เราสามารถอ้างอิงข้อมูลแต่ละฟิลด์ด้วยตัวแปร $1, $2 ... โดย $0 จะหมายถึงข้อมูลทั้งบรรทัด ตัวอย่างเช่น สมมติข้อมูลในแฟ้มที่เป็นฐานข้อมูลถูกบันทึกไว้ดังนี้

```
string11 string12 string13 … string1n
string21 string22 string23 … string2n
:        :        :  :      :
stringm1 stringm2 stringm3 … stringmn
```

การอ้างอิงถึงข้อมูลในแต่ละบรรทัด สมมติเป็นบรรทัดที่ 1 สามารถอ้างอิงได้เป็น

```
$1 = string11
$2 = string12
     :
$n = string1n
```

บรรทัดอื่น ๆ ก็สามารถอ้างอิงได้ในลักษณะเดียวกัน

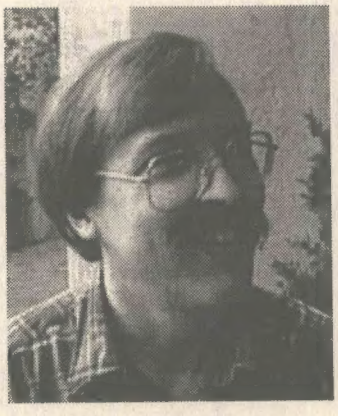
sed เป็น stream text editor ที่มีมากับระบบปฏิบัติการ UNIX ใช้สำหรับจัดการกับแฟ้มข้อมูลแบบตัวอักษรเช่นกัน สามารถสั่งงานได้จากทั้ง Command Line หรือระบุในเชลล์สคริปต์ก็ได้ ตัวอย่างการใช้งานได้แก่

```
sed 's/abc/xyz/' file1 > file1.out
```

คำสั่งข้างต้นใช้สำหรับเปลี่ยนคำ abc ในแฟ้ม file1 เป็น xyz ผลลัพธ์ที่ได้บันทึกลงในแฟ้ม file1.out
Perl มีการเลียนแบบการทำงานของทั้ง sed และ awk ข้างต้น และพัฒนาให้สามารถทำงานได้ดีกว่า

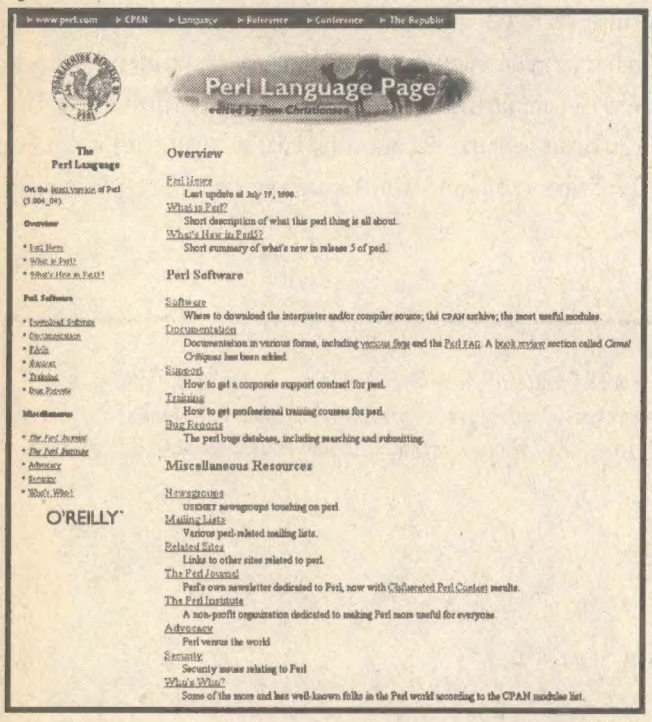
CGI คืออะไร?

CGI ย่อมาจากคำว่า Common Gateway Interface เป็นช่องทางที่ทำให้เราใช้ความสามารถของเว็บเซอร์ฟเวอร์ได้มากกว่าการที่จะทำหน้าที่ส่งเอกสาร HTML ไปยังผู้ใช้บริการเท่านั้น CGI ทำให้ผู้ใช้บริการสามารถเรียกใช้งานโปรแกรมที่อยู่บนเว็บเซอร์เวอร์ได้ ทำให้เว็บไซท์สามารถโต้ตอบกับผู้ใช้บริการได้ โดยทำการป้อนข้อมูลผ่านเอกสาร HTML ซึ่งจะถูกส่งผ่านไปยังโปรแกรมประยุกต์บนเซอร์ฟเวอร์ แล้วทำงานตามคำสั่งที่เขียนไว้ในโปรแกรม ผลลัพธ์ที่ได้จะแสดงผ่านเอกสาร HTML ให้ผู้ใช้บริการได้รับทราบ โปรแกรมที่เรียกใช้งานนี้จะเขียนขึ้นโดยภาษาใดก็ได้ที่เว็บเซอร์ฟเวอร์นั้น ๆ รองรับการทำงาน ไม่ว่าจะเป็นภาษา C, C++, Pascal, Perl หรือภาษาอื่นใด แต่ภาษาที่ได้รับความนิยมมากที่สุดในการใช้งานกับ CGI คือภาษา Perl



รูปที่ 1 โฉมหน้านาย Larry Wall ผู้ให้กำเนิด Perl

รูปที่ 2



JavaScript

Netscape Communication Corporation ได้เริ่มบรรจุภาษาสคริปต์ไว้ในเบราเซอร์ของตนเองตั้งแต่รุ่น 2.0 beta โดยให้ชื่อว่า "LiveScript" ต่อมาได้รับความร่วมมืออย่างใกล้ชิดจาก Sun Microsystem ร่วมกันปรับปรุงให้มีความสามารถเพิ่มขึ้น จนกระทั่งในเวอร์ชันจริงของ Netscape Navigator 2.0 ได้เปลี่ยนชื่อเป็น "JavaScript" นั่นเป็นจุดเริ่มต้นของภาษาสคริปต์ที่กำลังได้รับความนิยมมากที่สุดบนอินเทอร์เน็ตในปัจจุบัน

JavaScript ถูกสร้างขึ้นเพื่อใช้งานร่วมกับเอกสารแบบ HTML โดยเฉพาะ ถูกออกแบบมาเพื่อช่วยทำให้เอกสารแบบ HTML ซึ่งเดิมเป็นเอกสารนิ่ง ให้กลายเป็นเอกสารที่มีชีวิตขึ้น สามารถโต้ตอบกับผู้ใช้งานได้ ตัวสคริปต์ถูกฝังลงในเอกสาร HTML ซึ่งจะถูกส่งไปยังเครื่องไคลเอนต์เมื่อมีการเรียกดูเอกสารนั้น สคริปต์นี้จะถูกแปลโดยโปรแกรมที่ทำหน้าที่เป็นเว็บเบราเซอร์ ซึ่งรองรับการทำงานของ JavaScript อย่าง Netscape Navigator และ Internet Explorer จากนั้นจะทำงานตามคำสั่งที่เขียนไว้

โดยจุดประสงค์ของการสร้าง JavaScript นั้นต้องการให้ช่วยงานเล็ก ๆ น้อย ๆ ทางฝั่งของไคลเอนต์ โดยเฉพาะงานที่เกี่ยวข้องกับการติดต่อกับผู้ใช้ JavaScript ไม่ใช่ภาษาโปรแกรมอย่าง C หรือ C++ JavaScript สามารถบอกให้ทราบถึงเวลาในขณะนั้นบนเว็บเพจได้ทันทีด้วยคำสั่งเพียงบรรทัดเดียว JavaScript สามารถสร้าง PopUp Menu ได้ สามารถเปลี่ยนสีพื้นหลังของเอกสารเว็บได้ ถึงแม้สิ่งที่กล่าวมาทั้งหมดจะสามารถทำงานได้ด้วยการใช้ Perl เช่นเดียวกัน แต่ JavaScript ทำงานเหล่านี้ได้ง่ายดายกว่ามาก หากต้องการทำสิ่งเหล่านี้ด้วย Perl ผู้ใช้งานจำเป็นต้องเรียกติดต่อไปยังเว็บเซอร์ฟเวอร์ แล้วทำการโหลดเอกสารเว็บใหม่อีกครั้ง แต่การใช้ JavaScript สามารถทำงานได้ทันทีโดยไม่ต้องติดต่อกลับไปยังเซอร์ฟเวอร์อีก ซึ่งช่วยลดปริมาณข้อมูลที่วิ่งบนเครือข่ายเชื่อมโยงได้

งานบางประเภทสามารถทำงานได้โดย JavaScript เท่านั้น เช่นการทำ Image Swapping หรือการลากเมาส์ไปบนภาพแล้วทำให้ภาพที่ปรากฏเปลี่ยนไปจากเดิมทันทีทันใด ความสามารถในลักษณะนี้ Perl ทำไม่ได้

จุดเด่นของ JavaScript ประการหนึ่งคือการที่มันไม่จำเป็นต้องใช้ CGI พูดถึง CGI แล้วผู้ใช้อินเทอร์เน็ตหลายท่านคงไม่ชอบนัก เนื่องจากไม่ค่อยมีโอกาสได้ใช้งาน ยกเว้นจะมีเว็บเซอร์ฟเวอร์เป็นของตนเอง หรือได้รับอนุญาตให้ใช้งานจากผู้บริหารระบบ ชุมชนบนอินเทอร์เน็ตที่ได้รับความนิยมของชาวเน็ตส่วนใหญ่ไม่อนุญาตให้ใช้ CGI เนื่องจากยุ่งยากในการควบคุมการใช้งาน จึงดูเหมือนว่า JavaScript จะได้รับความนิยมและเป็นที่คุ้นเคยมากกว่า

JavaScript หลากหลายรุ่น

ทันทีที่ Microsoft ตัดสินใจบรรจุตัวแปลภาษา JavaScript ไว้ใน Internet Explorer 3.0 เป็นต้นมา JavaScript ก็กลายเป็นภาษามาตรฐานบนเว็บที่มีผู้ใช้งานมากที่สุดบนอินเทอร์เน็ตไปทันที ปัญหาใหญ่ของ JavaScript ก็คือการที่มีการใช้งานเว็บเบราเซอร์ที่แตกต่างกันมากมาย JavaScript ที่ Netscape และ Sun ร่วมกันพัฒนาจนถึงปัจจุบันมีถึง 3 รุ่น โดยรุ่นแรกเป็น JavaScript 1.0 ใช้งานใน Navigator 2.0 ต่อมาใน Navigator 3.0 ได้พัฒนาเป็น JavaScript 1.1 และรุ่นปัจจุบันที่ใช้กับ Navigator 4.0 เป็น JavaScript 1.2 สำหรับ Internet Explorer นั้น ทาง Microsoft พยายามที่จะรองรับการทำงานของ JavaScript 1.0 ใน Internet Explorer รุ่น 3.0 แต่เปลี่ยนชื่อเป็น "Jscript" แทน และใน Internet Explorer 4.0 ได้พยายามที่จะรองรับการทำงาน JavaScript 1.2 แต่ก็ยังคงรองรับได้ไม่ครบถ้วนสมบูรณ์ทั้งหมด การมี JavaScript หลายรุ่นทำให้เป็นปัญหาอย่างมากต่อผู้พัฒนาเอกสารเว็บ เนื่องจากคำสั่งของ JavaScript รุ่นใหม่จะไม่รองรับการทำงานในเบราเซอร์รุ่นเก่า ซึ่งจะฟ้องว่ามีข้อผิดพลาดขึ้น ในเบราเซอร์รุ่นใหม่ทั้ง Netscape Navigator และ Internet Explorer ซึ่งมีจำนวนผู้ใช้งานพอ ๆ กันในปัจจุบัน ก็ยังมีหลายสิ่งหลายอย่างที่เกี่ยวกับ JavaScript ที่แตกต่างกันอยู่ไม่น้อย

Server Side JavaScript

โดยปกติการกล่าวถึง JavaScript จะหมายถึงสคริปต์ที่ฝังในเอกสาร HTML และทำงานโดยเบราเซอร์ทางฝั่งไคลเอนต์ ในความเป็นจริง JavaScript สามารถทำงานบนฝั่งเซอร์ฟเวอร์ได้ด้วยเช่นกัน โดย Netscape Communication ได้พัฒนา LiveWire เพื่อใช้ในการบริหารเว็บเซอร์ฟเวอร์ขึ้น ซึ่งจะมี LiveWire JavaScript Compiler สามารถใช้แปลคำสั่ง JavaScript เพื่อใช้งานบนเซอร์ฟเวอร์ได้ แต่กลับไม่ได้รับความ

นิยมมากนัก เนื่องจาก LiveWire ทำงานกับเซอร์ฟเวอร์ของ Netscape เท่านั้น ซึ่งสัดส่วนการใช้งานเว็บเซอร์ฟเวอร์ของ Netscape ทุกรุ่นทุกแบบมีน้อยมากบนอินเทอร์เน็ต JavaScript ที่กล่าวถึงโดยทั่วไป รวมถึงในบทความนี้ จะหมายถึง JavaScript ที่ทำงานบนไคลเอนต์

เปรียบเทียบคุณสมบัติของ JavaScript และ Perl

ตารางที่ 1 แสดงคุณสมบัติเปรียบเทียบระหว่าง JavaScript และ Perl ซึ่งแบ่งเป็น 10 หัวข้อ

ตารางที่ 1

หัวข้อ	Perl	JavaScript
1. การใช้ทรัพยากรของระบบ	ทรัพยากรของระบบในที่นี้หมายถึงทั้งฝั่งไคลเอนต์ เซอร์ฟเวอร์ รวมถึงระบบเครือข่ายเชื่อมโยงด้วย ในการใช้งาน Perl (CGI) จะทำงานบนฝั่งเซอร์ฟเวอร์ การโต้ตอบระหว่างไคลเอนต์และเซอร์ฟเวอร์จำเป็นต้องผ่านระบบเครือข่ายเชื่อมโยง ทำให้ปริมาณข้อมูลที่วิ่งไปมาบนเครือข่ายมีสูงขึ้นด้วย	ในการใช้ JavaScript ข้อมูลทั้งหมดจะถูกโหลดไปยังเครื่องไคลเอนต์แล้วจึงทำงาน จึงไม่มีความจำเป็นจะต้องทำการติดต่อกับเซอร์ฟเวอร์อีก ช่วยลดปริมาณข้อมูลที่วิ่งบนระบบเครือข่ายเชื่อมโยงลงได้
2. ความยากง่ายในการใช้งาน	การใช้งาน Perl ต้องเรียกใช้งานผ่าน CGI ผู้พัฒนาเอกสารเว็บนั้น ๆ จึงจำเป็นต้องได้รับสิทธิ หรือได้รับอนุญาตจากผู้บริหารเว็บไซท์ให้ใช้งาน CGI และภาษา Perl ได้ ผู้ที่ไม่มีเซอร์ฟเวอร์เป็นของตนเองแต่ใช้วิธีเช่าพื้นที่บนคอมพิวเตอร์เพื่อสร้างเว็บไซท์ หรือชุมชนบนอินเทอร์เน็ตที่มีบริการให้พื้นที่สร้างโฮมเพจฟรี ซึ่งกำลังเป็นที่นิยมอย่างมากในปัจจุบัน มักจะไม่อนุญาตให้ใช้งาน CGI และ Perl เนื่องจากการที่จะให้คนจำนวนมากสามารถใช้ CGI ได้ เป็นจุดอ่อนต่อการรักษาความปลอดภัยของระบบ และยุ่งยากในการดูแลอย่างมาก	การใช้ JavaScript ไม่จำเป็นต้องได้รับสิทธิพิเศษใด ๆ ผู้ที่สามารถเขียนเอกสาร HTML ได้ ก็สามารถใช้งานได้แล้ว
3. การเรียกใช้ไลบรารีภายนอก	ในการพัฒนาโปรแกรมประยุกต์ขึ้นใช้งาน เราจะพบว่ามีงานส่วนหนึ่งที่มีการใช้บ่อย ๆ และใช้กับโปรแกรมประยุกต์หลาย ๆ โปรแกรม โปรแกรมเมอร์จึงมักจะเขียนขึ้นเป็นโปรแกรมย่อย เพื่อเรียกใช้งานโดยโปรแกรมหลัก โปรแกรมย่อยเหล่านี้ บางส่วนมักจะสามารถนำไปใช้งานกับหลาย ๆ โปรแกรมได้ จึงมีการรวบรวมโปรแกรมย่อยในแฟ้มใดแฟ้มหนึ่งในลักษณะที่เป็นไลบรารีที่สามารถเรียกใช้งานได้จากโปรแกรมอื่น Perl สามารถเรียกใช้ไลบรารีโดยระบุคำสั่ง require 'ชื่อแฟ้ม' เช่น require 'cgi-lib.pl' เป็นการเรียกใช้ไลบรารีในแฟ้ม cgi-lib.pl ทำให้เราสามารถเรียกใช้โปรแกรมย่อยที่มีอยู่ในแฟ้มไลบรารีนี้ได้	สำหรับ JavaScript ในรุ่นแรก ๆ ไม่สามารถจะเรียกใช้ไลบรารีภายนอกได้ จนกระทั่งได้มีการเพิ่มคำสั่ง SRC ในแท็ก <SCRIPT> ทำให้สามารถเรียกใช้ไลบรารีได้โดยระบุ ดังนี้ <SCRIPT SRC= "JavaScriptLibrary.js"> </SCRIPT> โดยแฟ้มที่เป็นไลบรารีนี้ต้องเป็นแฟ้มตัวอักษรที่มีนามสกุลเป็น .js
4. ความเข้ากันได้ของระบบ	Perl ถูกติดตั้งใช้งานบนเครื่องที่ทำหน้าที่เป็นเว็บเซอร์ฟเวอร์ สคริปต์สั่งงานก็ทำงานบนเซอร์ฟเวอร์ หากสคริปต์สามารถทำงานได้ถูกต้อง ผู้ที่เรียกใช้งานก็จะใช้งานได้อย่างไม่มีปัญหาแต่อย่างใด เนื่องจากจะไม่ขึ้นกับเว็บเบราเซอร์ที่ใช้งาน	JavaScript ทำงานบนฝั่งไคลเอนต์ เบราเซอร์เป็นผู้ทำหน้าที่แปลคำสั่งในสคริปต์แล้วจึงทำงานตามคำสั่งนั้น ๆ การทำงานจึงขึ้นอยู่กับโปรแกรมที่ใช้เป็นเบราเซอร์ด้วย แม้แต่ Netscape Navigator และ Internet Explorer ซึ่งต่างก็รองรับการทำงานของ JavaScript ทั้งคู่ก็ตาม ก็ยังอาจมีปัญหาในการทำงานได้ เนื่องจากมีมาตรฐานหลายรุ่น ได้แก่ JavaScript 1.0, 1.1 และล่าสุดเป็น JavaScript 1.2 เรื่องนี้สร้างปัญหายุ่งยากให้กับผู้พัฒนาเว็บไซท์เป็นอย่างมาก ทางออกที่ทำได้คือการเขียนสคริปต์ให้ทำการตรวจสอบก่อนว่าเป็นเบราเซอร์ใดรุ่นใด แล้วเลือกใช้คำสั่งที่เบราเซอร์นั้นรู้จัก

ควรใช้ JavaScript เมื่อไร

- เมื่อต้องการลดการทำงานของเซอร์ฟเวอร์
- เมื่อต้องการลดปริมาณข้อมูลที่ต้องส่งผ่านระบบสื่อสัญญาณ
- เมื่อต้องการตรวจสอบความถูกต้องของข้อมูลในเอกสารก่อนที่จะส่งไปยังเซอร์ฟเวอร์
- เมื่อต้องการให้เว็บเพจมีชีวิต สามารถโต้ตอบกับผู้ใช้งานได้ทันที
- ฯลฯ

ตารางที่ 1 (ต่อ)

หัวข้อ	Perl	JavaScript
5. การซ่อน/ปิดบังต้นฉบับ	การใช้งาน Perl หากมีการติดตั้งระบบอย่างเหมาะสม ผู้ใช้งานไม่สามารถจะเห็นโปรแกรมต้นฉบับได้ ทำให้ Perl เก็บความลับเกี่ยวกับการทำงานของโปรแกรมได้ดีกว่า	JavaScript ถูกฝังอยู่ในเอกสาร HTML ซึ่งจะถูกโหลดเข้าสู่เครื่องของไคลเอนต์ทั้งหมดก่อนจึงจะทำงานได้ ทำให้ผู้ใช้งานสามารถอ่านต้นฉบับของโปรแกรมได้โดยเพียงแต่ใช้โปรแกรมประเภท HTML Editor หรือ Text Editor ทั่ว ๆ ไปก็ได้ จึงไม่สามารถซ่อนหรือปิดบังต้นฉบับของสคริปต์ได้
6. การเข้าถึงข้อมูลในฮาร์ดดิสก์	Perl มีคำสั่งในการอ่าน/เขียนข้อมูลลงในฮาร์ดดิสก์เช่นเดียวกับภาษาโปรแกรมอื่น ๆ ทำให้บันทึกข้อมูลต่าง ๆ ได้ตามต้องการ	JavaScript ที่ใช้ในเว็บเบราเซอร์ จะต้องคำนึงถึงเรื่องความปลอดภัยของผู้ใช้งานเป็นหลัก จึงถูกจำกัดความสามารถไม่ให้เข้าถึงข้อมูลในฮาร์ดดิสก์ได้
7. การตรวจสอบข้อมูลของเบราเซอร์ที่เรียกใช้งาน	ทั้ง 2 ภาษา สามารถใช้ตรวจสอบข้อมูลรายละเอียดของเว็บเบราเซอร์ที่เข้ามาใช้บริการของเว็บไซท์ได้ โดย Perl ใช้วิธีตรวจสอบผ่านทางตัวแปรสภาวะแวดล้อม (Environment Variable)	JavaScript มีฟังก์ชันในการเรียกดูรายละเอียดของผู้ใช้บริการได้เช่นเดียวกัน
8. ความยากง่ายในการเขียนโปรแกรม/สคริปต์	Perl ถูกออกแบบให้ทำงานกับเอกสารที่เป็นตัวอักษรโดยเฉพาะ มีคำสั่งสำหรับดำเนินการกับข้อความต่าง ๆ ทำให้เขียนโปรแกรมได้ง่ายและกระชับ	Netscape ต้องการให้ JavaScript เป็นภาษาที่ง่ายต่อการใช้งาน ผู้ที่ไม่ใช่โปรแกรมเมอร์ก็สามารถเรียนรู้คำสั่งและสามารถเขียนสคริปต์ใช้งานได้อย่างไม่ลำบากยากเย็นนัก
9. ความปลอดภัย	Perl ถูกติดตั้งใช้งานบนเซอร์ฟเวอร์ จึงไม่มีผลกับเครื่องไคลเอนต์ Perl เป็นภาษาที่ได้รับการพัฒนามาเป็นอย่างดี โดยคำนึงถึงเรื่องเสถียรภาพของโปรแกรม และนับได้ว่าเป็นภาษาที่ปลอดภัยภาษาหนึ่ง หากเขียนโปรแกรมด้วยความระมัดระวัง แม้จะมีจุดอ่อนอยู่บ้างที่ต้องนำต้นฉบับของโปรแกรมไปไว้ในจุดที่ผู้ใช้บริการเรียกใช้งานได้ แทนที่จะเป็นโปรแกรมที่ถูกคอมไพล์แล้ว ทำให้ผู้ใช้อาจแอบดูต้นฉบับของโปรแกรมเพื่อศึกษาดูการทำงานเพื่อหาช่องโหว่ได้ แต่การติดตั้งระบบอย่างเหมาะสม สามารถเพิ่มความปลอดภัยในเรื่องนี้ได้	JavaScript ทำงานบนไคลเอนต์ จึงไม่สามารถคุกคามความปลอดภัยของเซอร์เวอร์ได้ จะมีปัญหาก็เฉพาะกับผู้ใช้อินเทอร์เน็ตเท่านั้น ปัญหาที่พบกันบ่อย ๆ ก็คือช่องโหว่ของโปรแกรมที่ทำหน้าที่เป็นเบราเซอร์ ที่ทำให้สามารถเขียนสคริปต์ที่สร้างความเสียหายให้กับเครื่องไคลเอนต์ได้ อย่างไรก็ตาม ช่องโหว่เหล่านี้มักจะได้รับการแก้ไขจากเจ้าของโปรแกรม โดยการออกโปรแกรมรุ่นใหม่แทนทันทีที่พบจุดบกพร่องเสมอ
10. ความสามารถในการใช้พัฒนาโปรแกรมประยุกต์	Perl มีโครงสร้างการทำงานและคำสั่งต่าง ๆ ให้ใช้งานเช่นเดียวกับภาษาโปรแกรมที่ดีอย่าง C, C++ มีกัน บางอย่างถูกพัฒนาให้ทำงานได้ดีกว่า ทำให้สามารถใช้ในการพัฒนาโปรแกรมประยุกต์ขนาดใหญ่ที่มีความสลับซับซ้อนได้ คำสั่งอื่น ๆ ที่มีอยู่ในภาษาโปรแกรมที่ดีทำให้สามารถทำงานกับโปรแกรมขนาดใหญ่ที่มีความสลับซับซ้อน รวมถึงการทำงานร่วมกับระบบฐานข้อมูลได้เป็นอย่างดี	JavaScript ถูกออกแบบให้ทำหน้าที่ในการช่วยให้เอกสารเว็บมีความสามารถในการโต้ตอบกับผู้ใช้ได้ แม้จะสามารถจัดการกับระบบฐานข้อมูลได้บ้าง แต่ก็ไม่เหมาะจะนำมาใช้จัดการกับฐานข้อมูลขนาดใหญ่ นอกจากนี้ JavaScript ยังมีข้อจำกัดอื่นอีกมากมาย

ควรใช้ Perl เมื่อไร

- เมื่อต้องการพัฒนาโปรแกรมประยุกต์ขึ้นใช้งาน
- เมื่อมีเซอร์ฟเวอร์ที่ทรงพลัง และต้องการใช้ความสามารถของเซอร์ฟเวอร์ให้เต็มประสิทธิภาพ
- เมื่อไม่ต้องการยุ่งยากกับการแก้ไขปัญหาความไม่เข้ากันของเบราเซอร์ต่างรุ่น ต่างแบบ มากมาย
- เมื่อมีความจำเป็นต้องมีการอ่าน/เขียนข้อมูลลงในแฟ้มบันทึกข้อมูล
- ฯลฯ

บทสรุป

ทั้ง JavaScript และ Perl ต่างก็เป็นเครื่องมือที่ดีในการสร้างสรรค์งานที่เกี่ยวกับเอกสารเว็บ แม้ทั้งสองภาษาจะมีความสามารถหลายอย่างเหมือนกัน แต่ก็มีคุณสมบัติและความสามารถที่ต่างกันหลายประการ งานบางอย่างสามารถเลือกใช้ภาษาใดภาษาหนึ่งก็ได้ แต่บางอย่างทำได้เฉพาะภาษาใดภาษาหนึ่งเท่านั้น จึงเป็นหน้าที่ของผู้นำไปใช้งาน ที่จะเลือกใช้ให้เหมาะสมกับงานที่ต้องการ เพื่อประสิทธิภาพการใช้งานของระบบสูงสุด